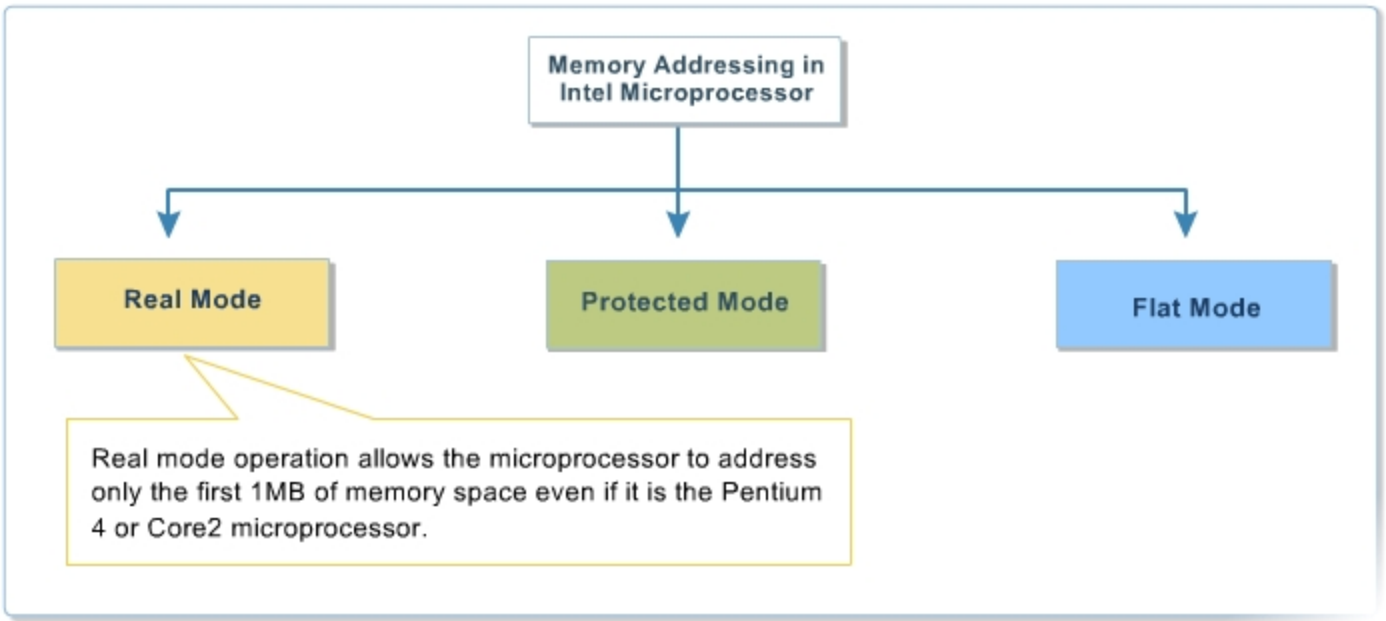
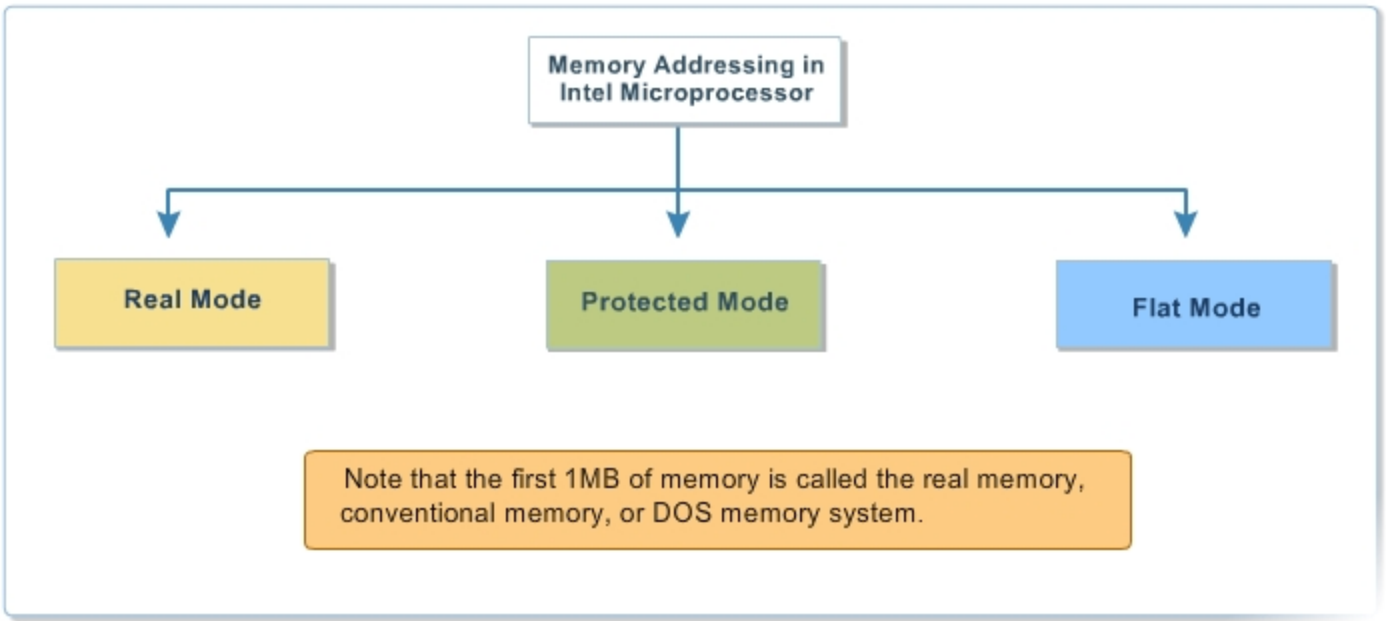
## Lesson 01: Introduction to Memory Modes

### Memory Addressing


Memory Addressing in Intel Microprocessor
Real Mode
Protected Mode
Flat Mode
Real mode operation allows the microprocessor to address only the first 1MB of memory space even if it is the Pentium 4 or Core2 microprocessor.

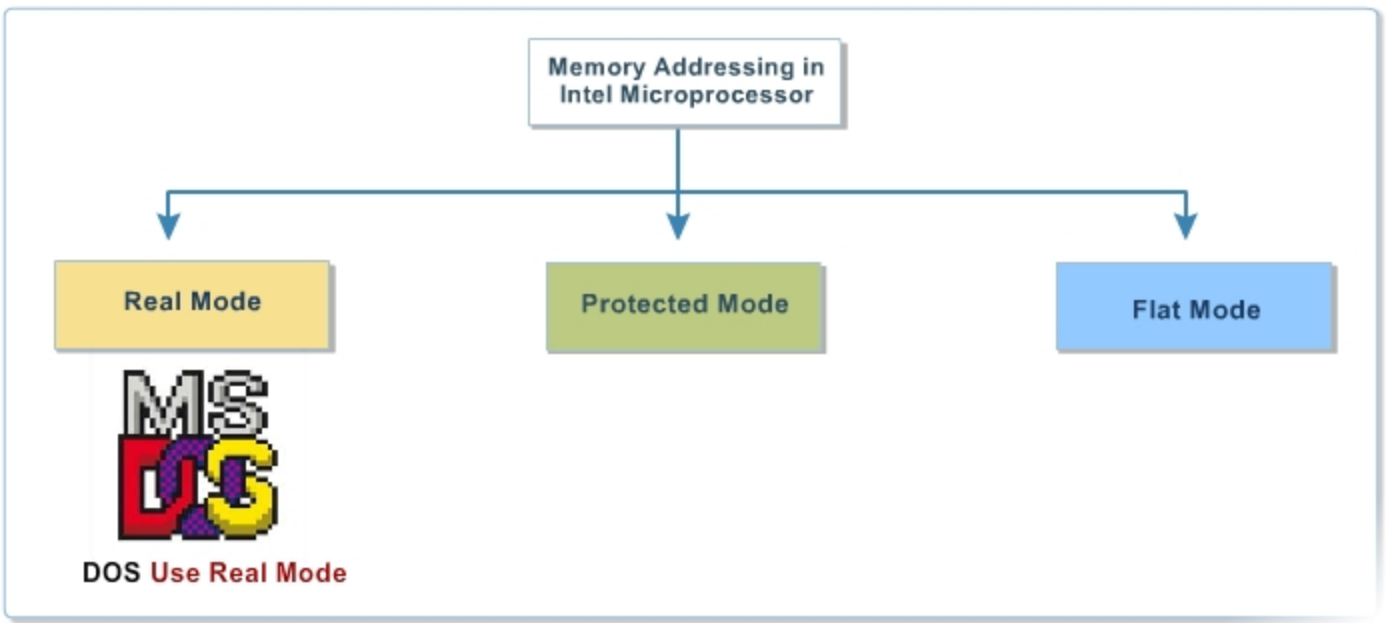
## Lesson 01: Introduction to Memory Modes

### Memory Addressing


Memory Addressing in Intel Microprocessor
Real Mode
Protected Mode
Flat Mode
Note that the first 1MB of memory is called the real memory, conventional memory, or DOS memory system.

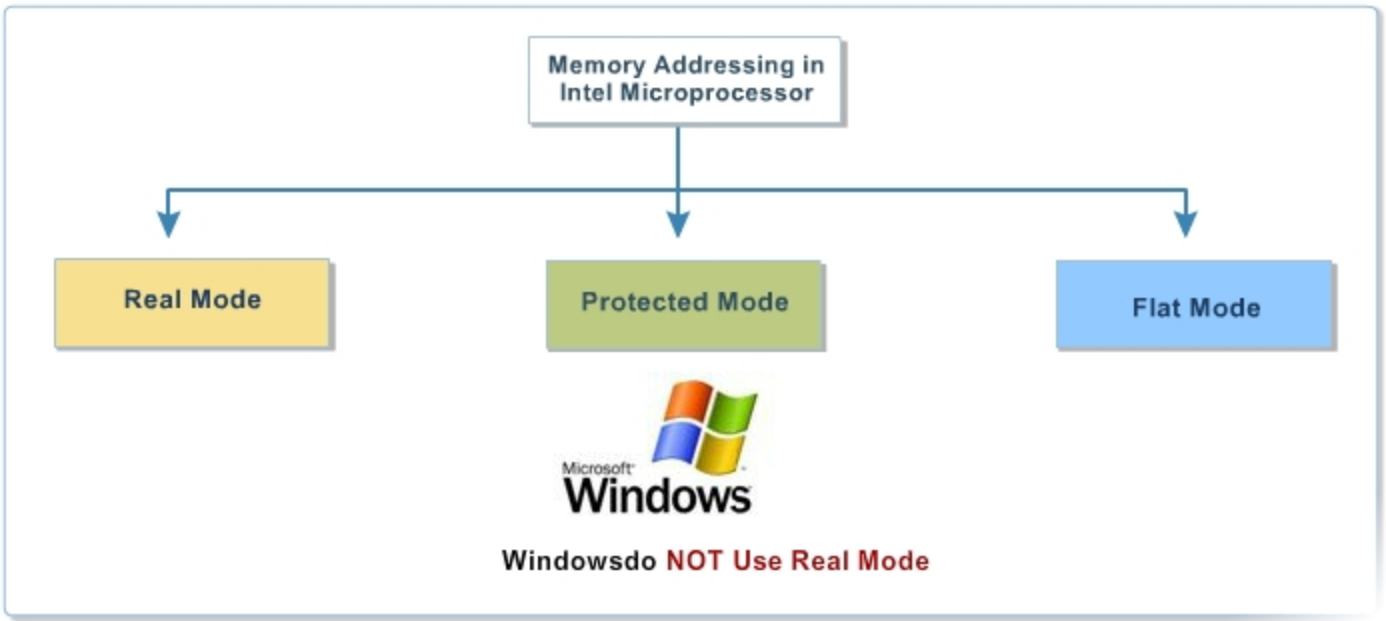
## Lesson 01: Introduction to Memory Modes

### Memory Addressing



## Lesson 01: Introduction to Memory Modes

### Memory Addressing

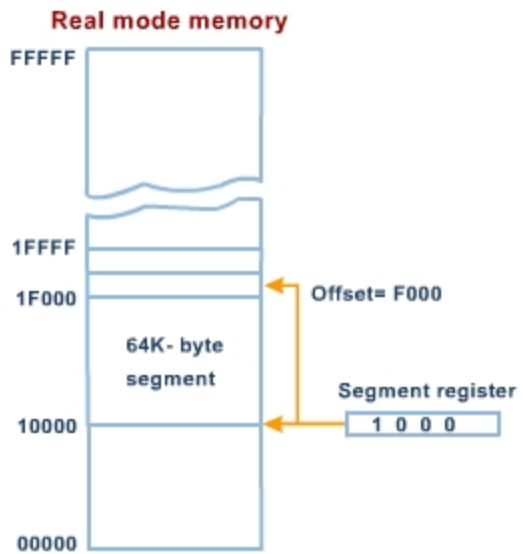


## Lesson 01: Introduction to Memory Modes

### Memory Addressing

#### Real Mode Memory Addressing

- In the real mode, a combination of a segment address and an offset address accesses a memory location.
- Such a scheme is called segment plus offset addressing scheme

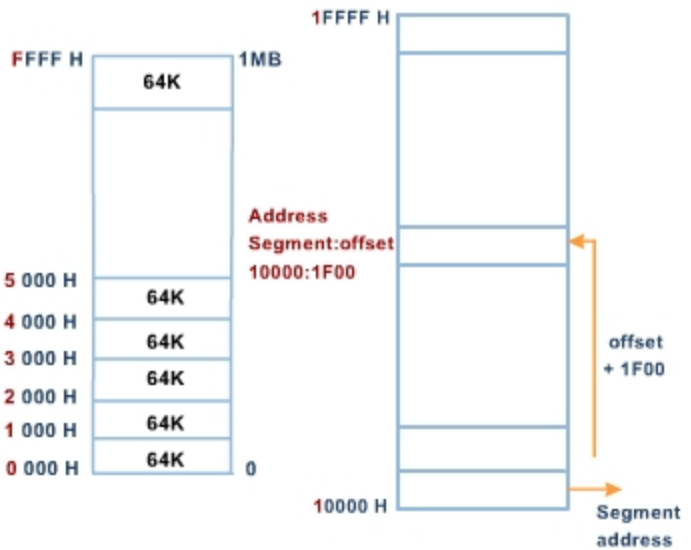


## Lesson 01: Introduction to Memory Modes

### Memory Addressing

#### Segments and Offsets

- The segment address, located within one of the segment registers, defines the beginning address of any 64K-byte memory segment.
- The offset address selects any location within the 64K byte memory segment.
- Segments in the real mode always have a length of 64K bytes.
- In the real mode, each segment register is internally appended with a 0H on its rightmost end. Therefore, real mode segments can begin only at a 16-byte boundary in the memory system.

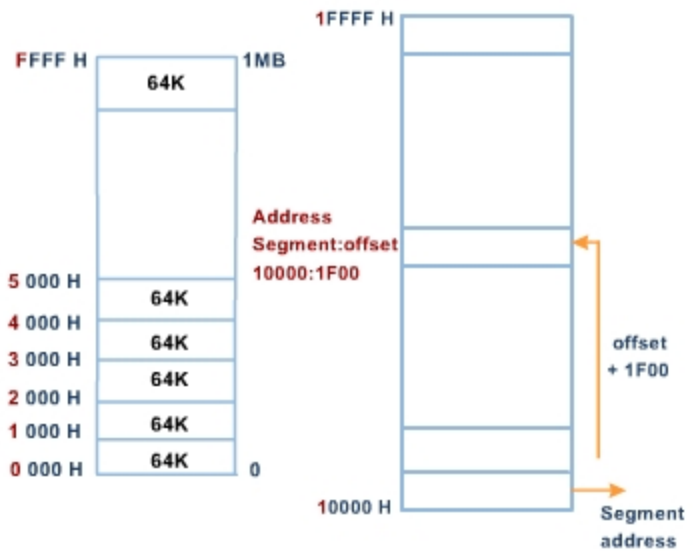


## Lesson 01: Introduction to Memory Modes

### Memory Addressing

#### Segments and Offsets

- The memory address in real mode of addressing is always described as **"Segment:Offset"**



## Lesson 01: Introduction to Memory Modes

### Memory Addressing

**Note....**

This l6-byte boundary is often called a paragraph. Because a real mode segment of memory is 64K in length, once the beginning address is known, the ending address is found by adding FFFFH.

## Lesson 01: Introduction to Memory Modes

### Segments and Offsets Example

**Example**

If the segment register contains 30000H, find the first and last address in that segment.

Instructions, you need to... Click to solution button.

Solution

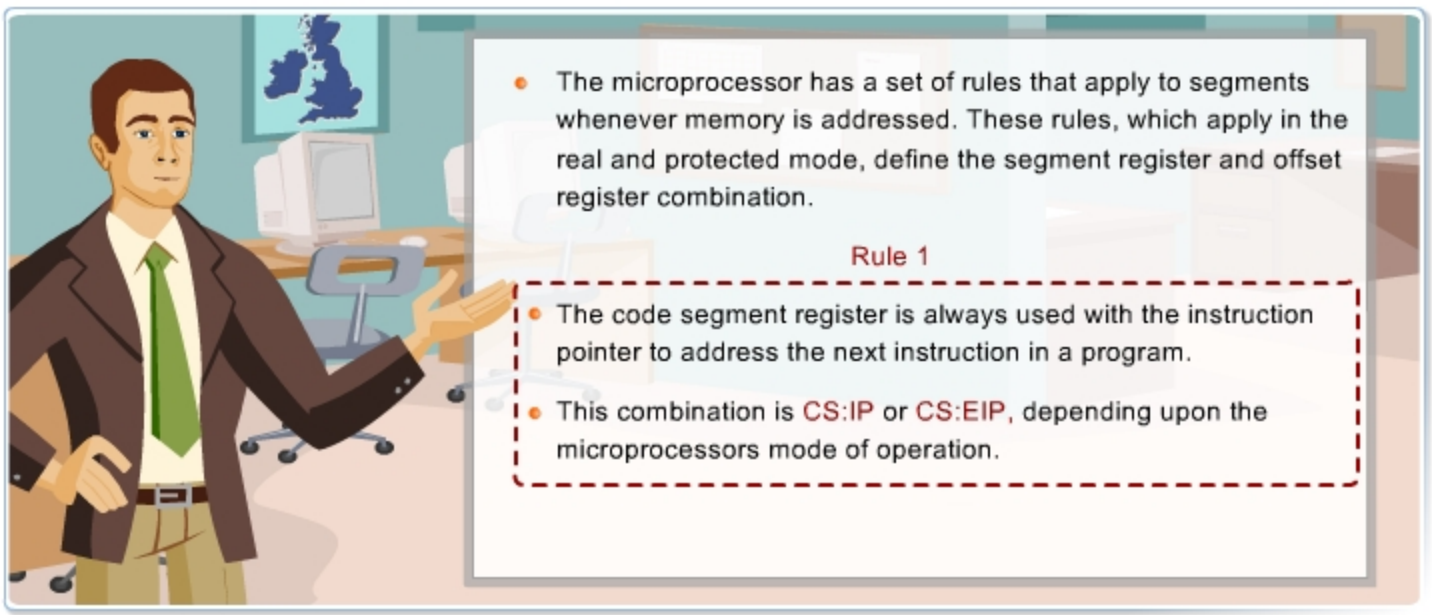
## Lesson 01: Introduction to Memory Modes

### Segments and Offsets Example

**Solution**

- The first address is that value of segment register appended by 0H. Therefore, the base address is 30000H.
- The last address is the base address plus FFFFH. Therefore, the last address is 3FFFFH.

## Default Segment and Offset Registration



- The microprocessor has a set of rules that apply to segments whenever memory is addressed. These rules, which apply in the real and protected mode, define the segment register and offset register combination.

Rule 1

- The code segment register is always used with the instruction pointer to address the next instruction in a program.
- This combination is CS:IP or CS:EIP, depending upon the microprocessors mode of operation.

## Default Segment and Offset Registration Example

**Example**

If CS=1400H and IP/EIP=1200H, find the memory location of the next instruction to fetch.

Instructions, you need to... Click to solution button.

Solution

## Lesson 01: Introduction to Memory Modes

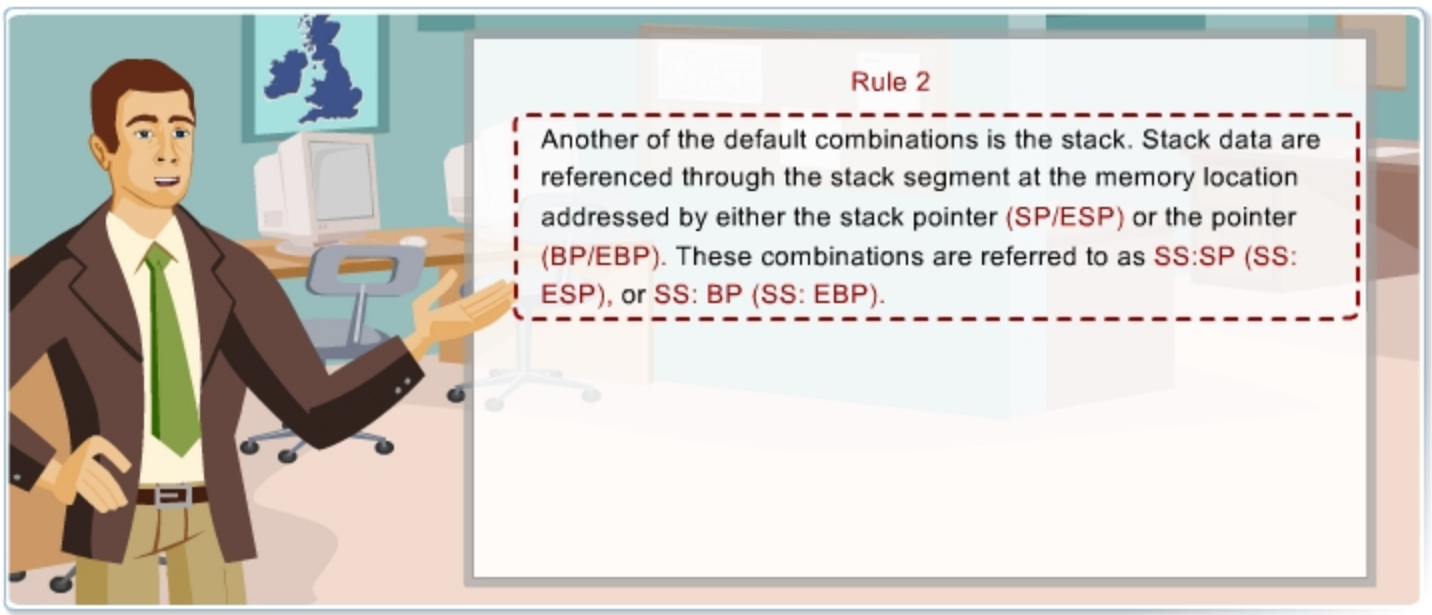
### Default Segment and Offset Registration Example

**Solution**

- Memory address of the location of the next instruction is CS appended by 0H + IP/EIP.
- Location is 15200H.

## Lesson 01: Introduction to Memory Modes

### Default Segment and Offset Registration


Rule 2
Another of the default combinations is the stack. Stack data are referenced through the stack segment at the memory location addressed by either the stack pointer (SP/ESP) or the pointer (BP/EBP). These combinations are referred to as SS:SP (SS: ESP), or SS: BP (SS: EBP).

## Lesson 01: Introduction to Memory Modes

### Default Segment and Offset Registration

**Example**

If SS=2000H and BP=3000H, which memory location is addressed for the stack segment memory.

## Lesson 01: Introduction to Memory Modes

### Default Segment and Offset Registration

**Solution**

- Memory address of the location of the stack memory segment is SS appended by 0H+ BP.
- The stack segment location is 23000H.

## Lesson 01: Introduction to Memory Modes

### Segments and Offsets Example

**Example**

An application program requires 1000H bytes of memory for its code, 190H bytes of memory for its data, and 200H bytes of memory for its stack.

It the DOS loader assigned the memory location 090F0H as base start of code segment. Draw a simple memory map for the code, data and stack memory segments.

Instructions, you need to... Click the solution button.

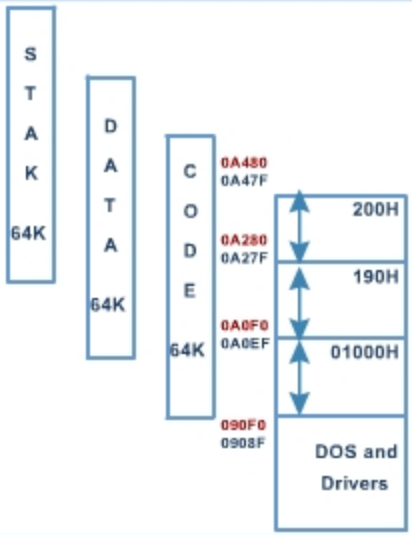
Solution

## Lesson 01: Introduction to Memory Modes

### Segments and Offsets Example

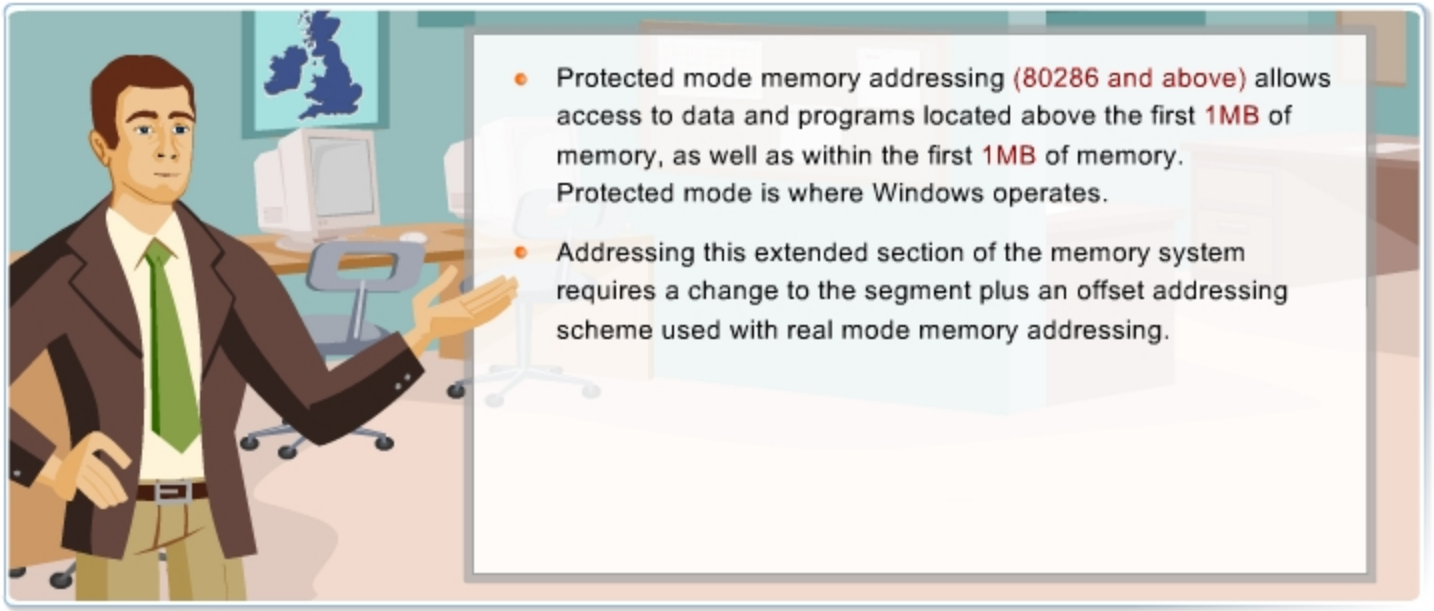
**Solution**

- In real memory mode, all segments are 64K length starting at:
- Code segment start location: 090F0H
- Data segment start location: 090F0H+1000H = 0A0F0H
- Stack segment start location: 0A0F0H+190H = 0A280H



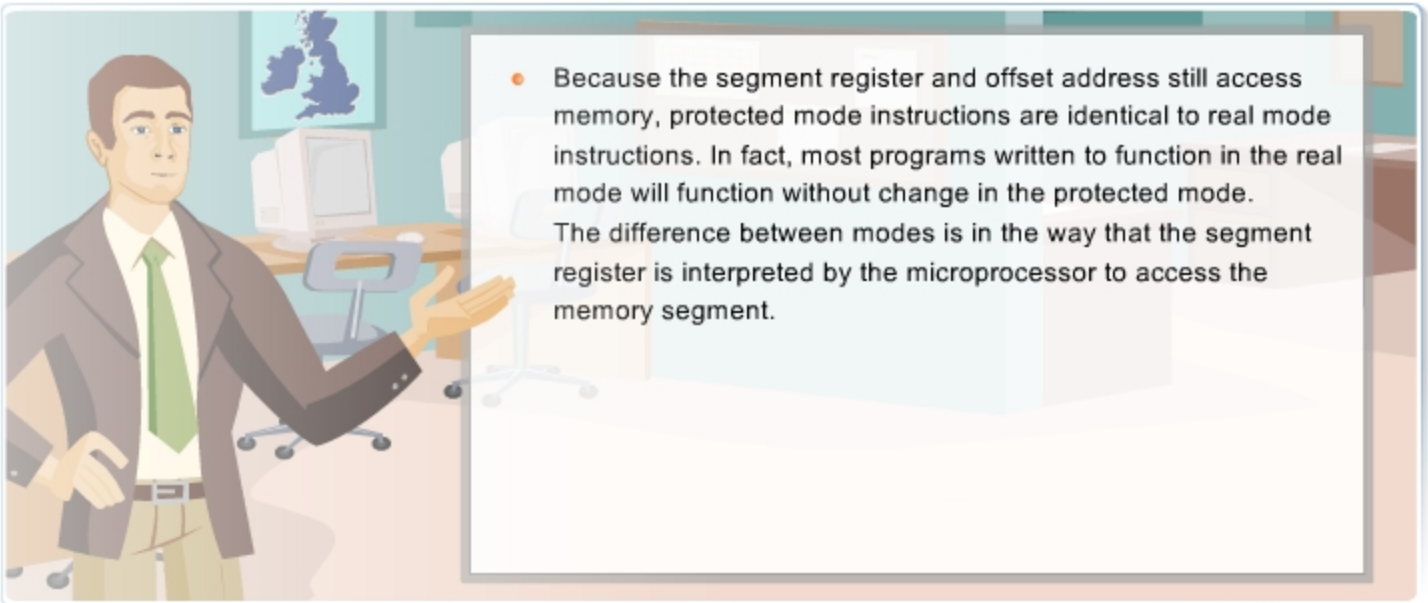
Lesson 01: Introduction to Memory Modes

## Protected Mode Memory Addressing


Protected mode memory addressing (80286 and above) allows access to data and programs located above the first 1MB of memory, as well as within the first 1MB of memory. Protected mode is where Windows operates.
Addressing this extended section of the memory system requires a change to the segment plus an offset addressing scheme used with real mode memory addressing.

Lesson 01: Introduction to Memory Modes

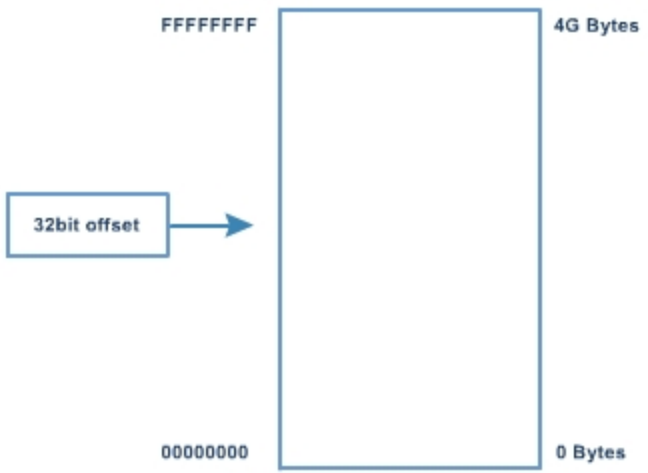
## Protected Mode Memory Addressing



- Because the segment register and offset address still access memory, protected mode instructions are identical to real mode instructions. In fact, most programs written to function in the real mode will function without change in the protected mode. The difference between modes is in the way that the segment register is interpreted by the microprocessor to access the memory segment.

## Protected Mode Memory Addressing

- A 32- bit offset address allows the microprocessor to access data within a segment that can be up to 4G bytes in length. Programs that are written for the 32- bit protected mode execute in the 64- bit mode of the Pentium 4.



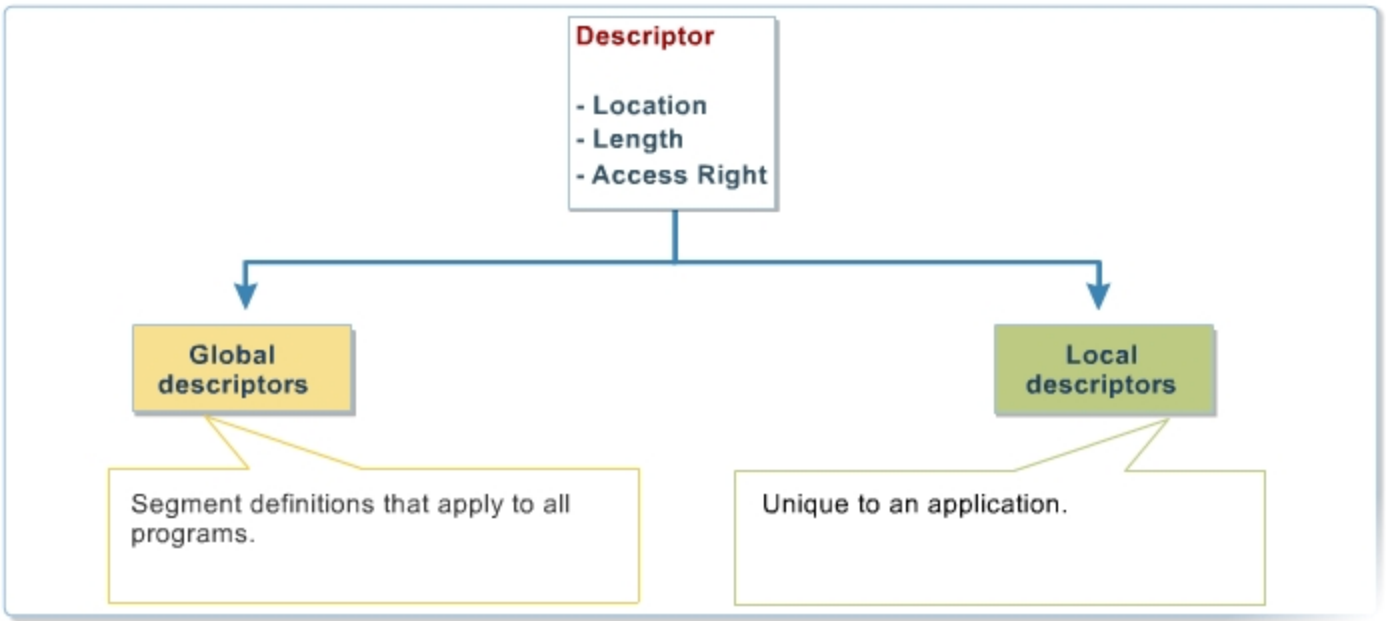
## Lesson 01: Introduction to Memory Modes

### Selectors and Descriptors

The selector, located in the segment register, selects one of 8192 descriptors from one of two tables of descriptors.

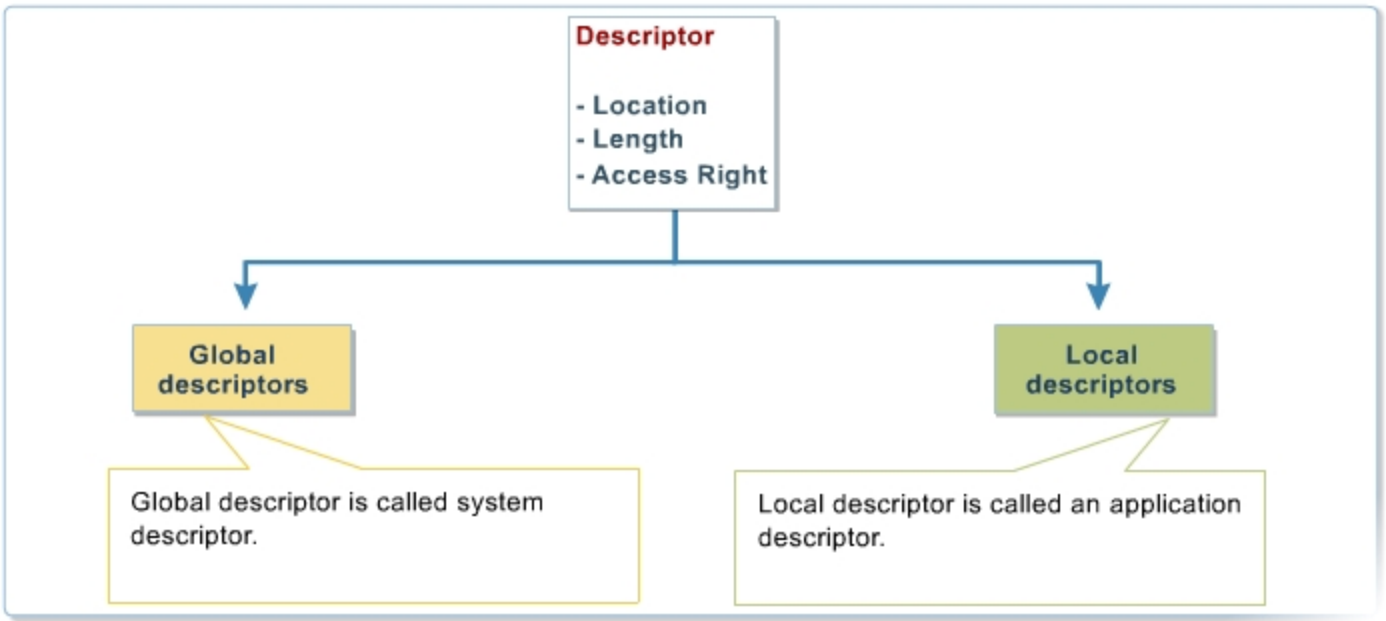
## Lesson 01: Introduction to Memory Modes

### Selectors and Descriptors



Lesson 01: Introduction to Memory Modes

## Selectors and Descriptors



## Selectors and Descriptors Example

**Example**

Calculate the maximum available memory for an application running under windows operating systems in protected mode.

Instructions, you need to... Click to solution button.

Solution

## Selectors and Descriptors Example

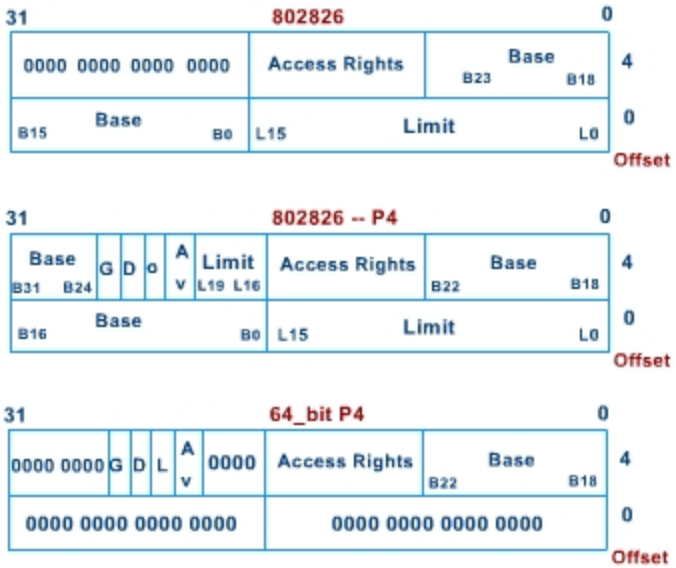
### Solution

- The application contains 2 descriptor tables. Each descriptor table contains 8192 (2000H or 8K) descriptors, so a total of 16,384 total descriptors are available to an application at any time.
- Because the descriptor describes a memory segment, this allows up to 16,384 memory segments to be described for each application.
- Since a memory segment can be up to 4GB in length, this means that an application could have access to 4G x 16,384 bytes of memory or 64TB.

## Lesson 01: Introduction to Memory Modes

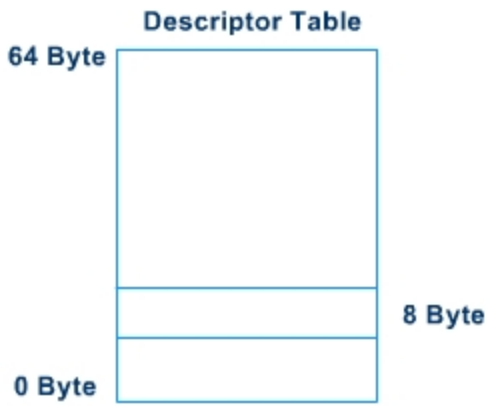
### Descriptor Format

The format of a descriptor for the 80286 through the Core2 is shown in the figure.



## Descriptor Format

- **Note:** That each descriptor is 8 bytes in length so the global and local descriptor tables are each a maximum of 64KB in length. Descriptors for the 80286 and the 80386 Core2 differ slightly, but the 80286 descriptor is upward-compatible.



## Lesson 01: Introduction to Memory Modes

### Descriptor Format

- The base address portion of the descriptor indicates the starting location of the memory segment. For the 80286 microprocessor the base address is a 24- bit address, so segments begin at any location in its 16MB of memory.
- **Note:** that the paragraph boundary limitation is removed in these microprocessors when operated in the protected mode so segments may begin at any address.
- The segment limit contains the last offset address found in a segment.

## Descriptor Format Example

### Example

If a segment begins at memory location F00000H and ends at location F000FFH, find the base address and the limit.

## Descriptor Format Example

### Solution

The base address is F00000H and the limit is FFH.
Notice that the 80286 has a 16- bit limit and the 80386 through the Pentium 4 has a 20- bit limit.

Therefore,

- (For the 80286) the base address is F00000H and the limit is 00FFH.
- (For the 80386) the base address is 00F00000H and the limit is 000FFH.

## Lesson 01: Introduction to Memory Modes

### Descriptor Format Example

**Note....**

Another feature found in the 80386 through the Pentium 4 descriptor that is not found in the 80286 descriptor: The G bit or granularity bit. If G=0, the limit specifies a segment limit of 00000H to FFFFFH. If G=1, the value of the limit is multiplied by 4KB (appended with FFFH).

## Lesson 01: Introduction to Memory Modes

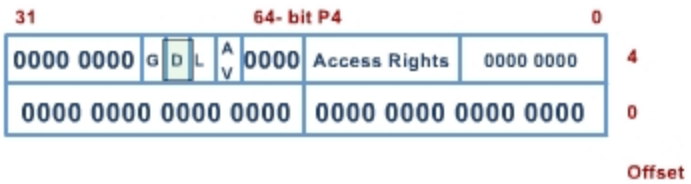
### Descriptor Format

- In the 64- bit descriptor, the L bit ( probably means large, but Intel calls it the 64- bit) selects 64-bit addresses in a Pentium 4 or Core2 with 64- bit extensions when L=1 and 32- bit compatibility mode when L=0.
- The AV bit, in the 80386 and above descriptor, is used by some operating systems to indicate that the segment is available (AV=1) or not available (AV=0).
- The D bit indicates how the 80386 through the Core2 instructions access register and memory data in the protected or real mode.

## Lesson 01: Introduction to Memory Modes

### Descriptor Format

- If D=0, the instructions are 16- bit instructions, compatible with the 8086 80286 microprocessors.



## Descriptor Format

### Example

Show the segment start and end when G bit is 0 and 1. Base address is 10000000H and the limit is 001FFH.

## Lesson 01: Introduction to Memory Modes

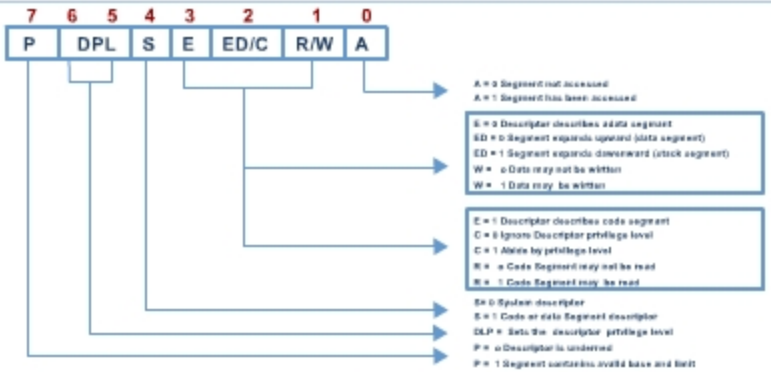
### Descriptor Format

**Solution**

- When G=0
  - Base = Start = 10000000H
  - End = Base + Limit = 10000000H + 001FFH = 100001FFH
- When G=1
  - Base = Start = 10000000H
  - End = Base + Limit = 10000000H + 001FFFFFH = 101FFFFFH

## Lesson 01: Introduction to Memory Modes

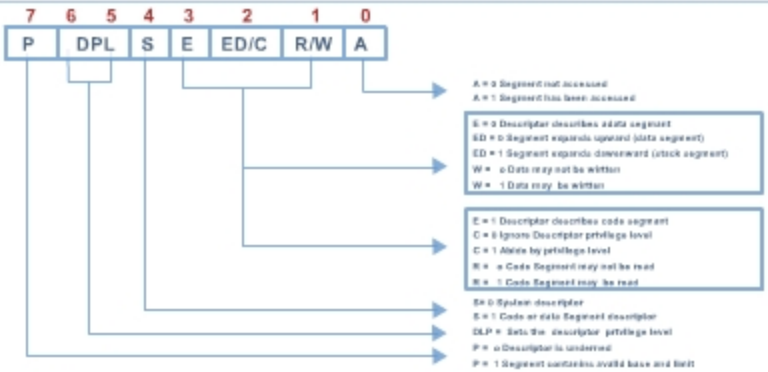
### Access Rights




- The access rights byte controls access to the protected mode segment. This byte describes how the segment functions in the system. The access rights byte allows complete control over the segment. If the segment is a data segment, the direction of growth is specified

## Lesson 01: Introduction to Memory Modes

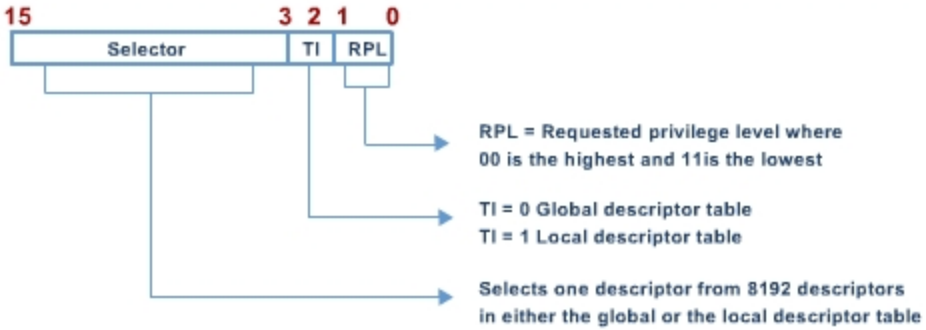
### Access Rights




- Descriptors are chosen from the descriptor table by the segment register.
  - The segment register contains a 13- bit selector field, a table selector bit, and a requested privilege level field.

## Lesson 01: Introduction to Memory Modes

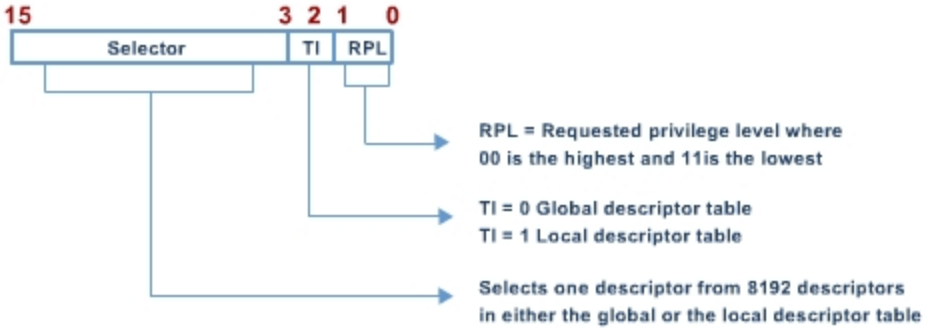
### Access Rights


15
3 2 1 0
Selector
TI
RPL
RPL = Requested privilege level where
00 is the highest and 11is the lowest
TI = 0 Global descriptor table
TI = 1 Local descriptor table
Selects one descriptor from 8192 descriptors
in either the global or the local descriptor table

- The TI bit selects either the global descriptor table (TI=0) or the local descriptor table (TI=1).
  - The requested privilege level (RPL) requests the access privilege level of a memory segment. The highest privilege level is 00 and the lowest is 11.

## Lesson 01: Introduction to Memory Modes

### Access Rights


15
3 2 1 0
Selector
TI
RPL
RPL = Requested privilege level where
00 is the highest and 11is the lowest
TI = 0 Global descriptor table
TI = 1 Local descriptor table
Selects one descriptor from 8192 descriptors
in either the global or the local descriptor table

- If the requested privilege level matches or is higher in priority than the privilege level set by the access rights byte, access is granted.

## Access Rights Example

**Example**

If DS = 0100H in a protected mode system, what is the requested privilege level?

## Access Rights Example

### Solution

The DS is determined from the right most byte of DS. Then the privilege level is 00H.

- Windows uses privilege level 00 (ring 0) for the kernel and driver programs and level 11 (ring 3) for applications. Windows does not use levels 01 or 10.

Lesson 01: Introduction to Memory Modes

## Access Rights Example

**Note....**

Note: If privilege levels are violated, the system normally indicates an application or privilege level violation.

## Lesson 01: Introduction to Memory Modes

### Summary

- All real mode memory addresses are a combination of a segment address plus an offset address.
- All instructions (code) are accessed by the combination of CS (segment address) plus IP or EIP (offset address).
- Data is normally referenced through a combination of the DS (data segment) and either an offset address or the contents of a register that contains the offset address.
- Protected mode descriptor contains a base address, limit, and access rights byte.
- The base address locates the starting address of the memory segment.
- The limit defines the last location of the segment.
- The access rights byte defines how the memory segment is accessed via a program.

## Lesson 01: Introduction to Memory Modes

### Activity

- Students may perform a discussion circle to develop some visual aid for memory map.